CLAIMS

What is claimed is:

[Note: Square-Bracketed **bold** ***and italicized*** **cross-referencing text** (e.g., **[*123*]**) is provided in the below claims as an aid for readability and for finding corresponding (but not limiting) examples of support in the specification. The bracketed text is not intended to add any limitation whatsoever to the claims and should be deleted in all legal interpretations of the claims and should also be deleted from the final published version of the claims.]

1. A contentions avoiding method [*Fig.2C*] for allowing an independently-clocked job requestor [*219"*] to request respective processing [*255*] of respectively requested jobs in an independently clocked, job processor [*253"*] where variable communication latencies [*103*] may exist between the job requestor and the job processor, said method being machine implemented and comprising:

(a) issuing [*232*] to the job requestor, a first time stamp [*GTSa*] representing a respectively scheduled, first time within a timing reference frame [*256'*] of the job processor at which a respective first job is to be performed;

(b) in response to receipt of the first time stamp, sending [*240"*] from the job requestor to the job processor, a combination of job payload data [*250.1*] and a second time stamp [*GTSb*] also representing the scheduled, first time;

(c) in response to receipt by the job processor of the combination [*240'*] of the job payload data and the second time stamp, storing [*241"*] the received job payload data in [*254*] the job processor; and

(d) causing the job processor to process [*245.1*] the stored payload data when a time corresponding to the second time stamp [*GTSb*] occurs within the timing reference frame [*256'*] of the job processor.

2. A rates equalizing method [*Fig.2A,7A*] for equalizing effective data exchange rates between an independently-clocked job requestor [*219*] and an independently-clocked, job processor [*253*],

where the job requestor sends to the job processor, requests [*131a*] for respective processings [*155*] of job payloads [*131b*] which are to be later supplied from the job requestor to the job processor,

where in response to at least some of said sent requests, the job processor sends to the job requestor corresponding grants [*132a*], and

where variable latencies [*103*] may exist for communications between the job requestor and the job processor,

said rates equalizing method being machine implemented and comprising:

(a) statically constraining [*206*] the average grant-issuing rate of the job processor to be no greater than a tolerable average grant-receiving rate which is tolerable by the job requestor over a relatively long span of time; and

(b) dynamically constraining [*146*] the request-issuing rate of the job requestor to be no greater than a tolerable average request-receiving rate which is tolerable by the job processor over said relatively long span of time.

3. A data sets re-ordering method [*Fig.1B,6*] for returning to a pre-established order [*142*] and on-the-fly, data sets which are transmitted from different source units [*119,129*], over a distributed interconnect [*103*] and processed within a distributed processing fabric [*105*] and after said processing, transmitted to a common destination unit [*680*], where different parts of the distributed interconnect are subject to respectively different or variable communication latencies, said re-ordering method being machine implemented and comprising:

(a) associating with each transmitted data set [*147,148*] a respective source indicator [*526*], a logical sequence indicator [*525*], and a process scheduling time stamp [*522*];

(b) within the distributed processing fabric, processing each transmitted data set at a respective local time [*291*] of the distributed processing fabric, where the respective local time for processing is identified by the process scheduling time stamp [*GTSb*] of the respective, transmitted data set; and

(c) as source and sequence associated ones of the transmitted data sets [*684-691*] arrive sequentially at the common destination unit, positionally swapping [*686*] the sequentially arriving data sets on-the-fly and in accordance with their respective source and sequence indicators so as to bring data sets of respective sources into relative sequence with one another as specified by the sequence indicators of respective source units.

4. A distributed payload processing system [*100,700*] comprising:

(a) a first circuit supporter [*102*] having a respective first, independent clock [*117*], where the first circuit supporter is adapted to support a first set of one or more of job requestor units [*ZINCs,119,129*] and job processor units [*ZESTs,151,152*], where said units of the first set are clocked by the first independent clock;

(b) a second circuit supporter [*106*] having a respective second, independent clock [*157*], where the second circuit supporter is adapted to support a second set of one or more of job requestor units [*ZINCs,119,129*] and job processor units [*ZESTs,151,152*], where said units of the second set are clocked by the second independent clock;

(c) an interconnect [*103*] operatively coupling the first circuit supporter [*102*] with the second circuit supporter [*106*] such that job requestor units [*ZINCs,119,129*] of one of the first and second circuit supporters can send to one or more job processors in the other of said circuit supporters, requests [*131a*] for respective processings [*155*] of job payloads [*131b*] which are to be later supplied from the corresponding job requestors to the respective job processors,

where in response to at least some of said sent requests, the respective job processors send to the corresponding job requestors corresponding grants [*132a*], and

where variable latencies [*103*] may exist within the interconnect [*103*] for communications carried over the interconnect between the job requestors and the job processors, and

(c.1) said grants [*132a*] that are sent over the interconnect include respective time stamps [*GTSa*] representing respectively scheduled time slots within respective timing reference frames [*256'*] of the respective job processors at which respective ones of requested processing jobs are to be performed in the grant-giving job processors.

5. A scheduling support subsystem [*254,751-752*] for enabling scheduling [*252,740*] of payload processing actions [*255,754*] in a request/grant system [*200,700*] where effective latency between grant release [*156,264,711*] and completed arrival [*154*] in a payload processing unit [*253,750*] of a corresponding payload [*250b,719*] can vary as a function of one or more latency factors in the group consisting of: (1) respective locations of payload source and payload processing units, (2) temperature, (3) time, (4) clock frequency and/or clock phase differences [*CLK3-CLK2*], (5) interconnect link lengths [*705*] between respective points [*156*] of grant release and respective points [*154*] of payload arrival, and (6) local synchronization window lengths and/or phases [*206*] at the respective locations of payload source and payload processing units; where the scheduling support subsystem comprises:

(a) a grant stamper [*745,230*] for stamping released grants with respective time markers [*GTSa,STSa*] representing corresponding time points in a time frame of the payload processing unit, said represented time points being when the payloads corresponding to the released grants are to be processed by the payload processing unit; and

(b) latency fixing means for fixing latency between grant release [*156*] and time of payload processing [*154*] in the payload processing unit essentially to a predefined round-trip value [T_{GS}].

6. The scheduling support subsystem of Claim 5 wherein:

(b.1) the predefined round-trip value [T_{GS}] is at least greater than the average of non anomalous ones of said effective latencies between the grant releases and the completed arrivals of the corresponding payloads in the system.

7. The scheduling support subsystem of Claim 5 wherein:

(b.1) the predefined round-trip value [T_{GS}] is at least as great as the maximum, non-infinite and non anomalous one of said effective latencies between the grant releases and the completed arrivals of the corresponding payloads in the system.

8. The scheduling support subsystem of Claim 5 wherein the latency fixing means includes:

(b.1) an alignment queue [*254,751*] provided in the payload processing unit for delaying the processing of received payloads to respective local times corresponding with payload stamps [*GTSb,STSb*] accompanying the received payloads, where the payload stamps correspond to the time markers stamped by the grant stamper.

9. The scheduling support subsystem of Claim 8 wherein the latency fixing means includes:

(b.2) delay adjusting means [*233,744*] for adjusting said essentially fixed latency between grant release and time of payload processing in the payload processing unit.

10. The scheduling support subsystem of Claim 5 wherein the latency fixing means includes:

(b.1) a local time defining means [*235,755*] provided in the payload processing unit for defining when in a local time frame [*CLK3,CLK-D*] of the payload processing unit said corresponding time points occur, where the time points correspond to the time markers [*GTSa,STSa*] stamped by the grant stamper.

11. The scheduling support subsystem of Claim 10 wherein:

(a.1) the grant stamper includes a first sequencer [*230,GTC*] which sequences through a first wraparound sequence of markers that respectively represent said corresponding time points in the time frame of the payload processing unit; and

(b.1a) the local time defining means includes a second sequencer [*233,235*], that is operatively coupled to the first sequencer and that sequences through the first wraparound sequence of markers while lagging behind said sequencing of the first sequencer by a configurable lag factor [*RTA*], where said configurable lag factor can vary at least as much as can; for respective grant-carrying links [*132a*] and payload-carrying links [*131b*] of the system, the link-related latencies between the grant releases [*156*] and the completed arrivals [*154*] of the corresponding payloads on respective interconnect links in the system so that, given an actual latency with its corresponding variance for a corresponding grant-carrying link [*132a*] and a corresponding payload-carrying link [*131b*], the sum of time [T_{SAQ}] spent sitting in an alignment queue by a given, arrived payload and of the actual latency experienced by that payload defines said, essentially fixed latency value [T_{GS}=*RTA*].

12. The scheduling support subsystem of Claim 11 wherein:

(a.1a) the first sequencer [*745,STC*] is clocked [*747*] independently of the second sequencer [*755,PTC*].

13. The scheduling support subsystem of Claim 12 wherein:

(b.1b) the variable lag factor [*STC-PTC*] between the first and second sequencers is determined from test transmissions [*744a,b*] sent from a respective locale [*740*] of one of the first and second sequencers to a respective other locale [*750*] of another of the first and second sequencers by way of different ones of interconnect link lengths [*711,721*] provided between a respective point [*756*] of grant release and respective points [*751,752*] of payload arrival.

14. A clock-tree free, scalable system [*100,700*] for moving payload signals [*131b,719*] from respectively-clocked source circuits [*119,129,710*] and through an independently-clocked, payload-processing layer [*105,750*] for subsequent delivery to respectively-clocked destination circuits [*209,760*] without need for a clock tree extending between the payload-processing layer and either of the source or destination circuits, where said destination circuits may include said source circuits [*119,129*] or other respectively-clocked ones of the destination circuits; where a source-to-processing-to-destination interconnect [*103,704*] is provided for transmitting payload signals, request and corresponding grant signals, and other signals between the source circuits, the payload-processing layer, and the destination circuits, and where said interconnect can exhibit different signals transmission latencies between various ones of the source circuits, different processing parts of the payload-processing layer, and the destination circuits; said scalable system [*100,700*] comprising:

(a) grant stamping units [*252,745*] for stamping respective grant signals [*264',711*] each with a grant time stamp [*GTSa,STSa*] that indicates when, within a respective time frame [*257,757*] of a corresponding, processing part [*253,750*] of the payload-processing layer, processing is to take place for a payload signal corresponding to the stamped grant signal;

(b) alignment queues [*254.1,254.2,751*] provided adjacent to corresponding ones of the payload-processing parts [*253*] of the payload-

processing layer, for storing respective payload signals received by the payload-processing parts prior to processing times indicated by respective ones of the grant time stamps; and

(c) process coordinating means [*249,753*], operatively coupled to the alignment queues and responsive to the grant time stamps or stamp signals [*GTSb,STSb*] derived therefrom, for controlling which of the payload signals stored in the alignment queues will be processed by which of the payload-processing parts and when, in accordance with the processing times indicated by respective ones of the grant time stamps or said stamp signals derived therefrom.

15. The scalable system [*100,700*] of Claim 14 wherein each grant stamping unit includes:

(a.1) a grant time counter [*230*] that sequences through a wrap-around series of states that represent grantable processing time slots in a corresponding payload-processing part [*253*], where the grant time counter is clocked by a local timing generator [*256*] of the grant stamping unit; and

(a.2) a grant source queue [*249c*] for storing grant identification information [*232c,237c*] indicating for respective ones of the grantable processing time slots at least whether a valid grant signal was issued with a corresponding grant time stamp for that grantable processing time slot or at least which of said plural align queues [*255.1-255.N*] a corresponding payload signal is to be acquired from for processing during that grantable processing time slot.

16. The scalable system [*100,700*] of Claim 14 wherein for each of the alignment queues said scalable system [*100,700*] includes:

(b.1) popping means [*290,292*] for popping from a respective one of the alignment queues, at or just before the occurrence of a granted, processing time slot, the corresponding payload signal [*240.1"*] of that granted, processing time slot so that the popped payload signal can be duly

processed during the granted, processing time slot by the corresponding payload-processing part [*255*].

17. The scalable system [*100,700*] of Claim 16 wherein for each of the alignment queues said scalable system [*100,700*] further includes:

(b.2) pushing means [*239*] for pushing into an unused storage space [*240.9"*] of a respective one of the alignment queues, and before the occurrence of a granted, processing time slot, the corresponding payload signal [*240.1"*] of that granted, processing time slot so that the pushed payload signal can be later popped and duly processed during the granted, processing time slot by the corresponding payload-processing part [*255*].

18. The scalable system [*100,700*] of Claim 17 wherein each pushing means includes:

(b.2a) null dropping means [*227"*] which excludes from said pushing into unused storage space, data of a received payload signal if the received payload signal is indicated to be invalid.

19. The scalable system [*100,700*] of Claim 17 wherein each of the alignment queues said scalable system [*100,700*] further includes:

(b.3) latency determining means [*294*], coupled to the pushing means and to the popping means, for determining what relative latency [*295*] separates currently popped payloads from currently pushed payloads and for outputting a signal [*G-S_gap*] representing that relative latency.

20. The scalable system [*100,700*] of Claim 19 and further comprising:

(b.4) latency evaluating means [*295*], coupled to a plurality of said latency determining means of a respective plurality of the alignment queues, the latency evaluating means being for evaluating the relative latency output signals of said plurality of latency determining means and for determining

whether corrective action needs to be taken if all or a subset of said relative latency output signals indicate respective relative latencies outside of predefined bounds established for such relative latencies.

21. The scalable system [*100,700*] of Claim 14 wherein said payload-processing parts each includes a payload switching matrix [*255'*].

22. A distributable set [*105*] of locally synchronous, payload processing units [*151-15m*] where each said payload processing unit comprises:

(a) a plurality of payload ingress ports [*224*];

(b) a plurality of payload egress ports [*265*];

(c) local timing means [*256*] for defining local processing windows [*206*];

(d) a payloads processing module [*255*], operatively coupled to the local timing means, for processing respective payloads during respective and pre-scheduled ones of the local processing windows [*206*];

(e) a payloads aligning module [*254*], operatively coupled to the payload ingress ports and to the payloads processing module, for receiving respective payloads from the ingress ports and for time-aligning the received payloads for delivery at their respective and pre-scheduled ones of the local processing windows, to the payloads processing module for processing by the payloads processing module; and

(f) a post-process output module [*612,615*], operatively coupled to the payload egress ports and to the payloads processing module, for receiving processed payloads from the payloads processing module, and for packaging the processed payloads for output by way of the payload egress ports [*645-649*].

23. The distributable set [*105*] of Claim 22 wherein:

(a.1) said payload ingress ports [*224*] are asynchronous ports structured to receive asynchronous ingress signals containing respective ones of ingressing payloads, and each said payload processing unit [*253*] further comprises:

(g) for each ingress port, a corresponding clock recovery mechanism [*245*] for recovering an embedded clock signal [*245o*] from the respective asynchronous ingress signals of the corresponding ingress port;

(h) for each ingress port, a corresponding sampling means [*244*], operatively coupled to the corresponding clock recovery mechanism and to the corresponding ingress port, for sampling data bits of the respective asynchronous ingress signals of the corresponding ingress port;

(i) for each ingress port, a corresponding re-sampling means [*247*], operatively coupled to the corresponding sampling means [*244*] and to the local timing means [*256*], for re-aligning the sampled data bits to the local processing windows [*206*] of the payload processing unit [*253*].

24. The distributable set [*105*] of Claim 22 wherein:

(a.1) said payload egress ports [*265*] are asynchronous ports structured to output asynchronous egress signals containing respective ones of processed payloads, and each said payload processing unit [*253*] further comprises:

(g) for each egress port, a corresponding code converter [*612*] for converting the data of respective ones of processed payloads into a format that includes an embedded and recoverable clock signal; and

(h) for each egress port, a corresponding code inserter [*615*] for inserting one or both of synchronization code signals and idle code signals into the asynchronous egress signals output by the respective egress port.

25. The distributable set [*105*] of Claim 24 wherein each said payload processing unit [*253*] further comprises:

(i) for each egress port, a corresponding ECC inserter [*612*] for inserting an error checking and correcting code [*545*] that covers at least a corresponding one of the processed payloads output by the corresponding egress port.

26. A payload processing unit [*253*] comprising:

(a) a plurality of ingress ports [*224*] for receiving ingressing payload and other signals;

(b) a plurality of egress ports [*265*] for output egressing payload and other signals;

(c) local timing means [*256*] for defining local processing windows [*206*] of the payload processing unit;

(d) a payloads processing module [*255*], operatively coupled to the local timing means, for processing respective payloads during respective and pre-scheduled ones of the local processing windows [*206*];

(e) a payloads aligning module [*254*], operatively coupled to the ingress ports and to the payloads processing module, for receiving respective ingressing payloads from the ingress ports and for time-aligning the received payloads for delivery at their respective and pre-scheduled ones of the local processing windows, to the payloads processing module for processing by the payloads processing module; and

(f) a post-process output module [*612,615*], operatively coupled to the egress ports and to the payloads processing module, for receiving processed payloads from the payloads processing module, and for packaging the processed payloads for output by way of the egress ports [*645-649*].

27. The payload processing unit [*253*] of Claim 26 wherein:

(a.1) said ingress ports [*224*] are asynchronous ports structured to receive asynchronous ingress signals containing respective ones of ingressing payloads, and the payload processing unit [*253*] further comprises:

(g) for each ingress port, a corresponding clock recovery mechanism [*245*] for recovering an embedded clock signal [*245o*] from the respective asynchronous ingress signals of the corresponding ingress port;

(h) for each ingress port, a corresponding sampling means [*244*], operatively coupled to the corresponding clock recovery mechanism and to the corresponding ingress port, for sampling data bits of the respective asynchronous ingress signals of the corresponding ingress port;

(i) a re-sampling means [*247*], operatively coupled to the sampling means [*244*] of at least one ingress port and to the local timing means [*256*], for re-aligning the sampled data bits of the at least one ingress port to the local processing windows [*206*] of the payload processing unit [*253*].

28. The payload processing unit [*253*] of Claim 26 wherein:

(a.1) said egress ports [*265*] are asynchronous ports structured to output asynchronous egress signals containing respective ones of processed payloads, and the payload processing unit [*253*] further comprises:

(g) an egress traffic, code converter [*612*] for converting the data of respective ones of processed payloads into a format that includes an embedded and recoverable clock signal; and

(h) an egress traffic, code inserter [*615*] for inserting one or both of synchronization code signals and idle code signals into the asynchronous egress signals output by respective ones of the egress ports.

29. The payload processing unit [*253*] of Claim 28 wherein the payload processing unit [*253*] further comprises:

(i) an ECC inserter [*612*] for inserting an error checking and correcting code [*545*] that covers at least a corresponding one of the processed payloads output by a corresponding egress port.

30. The payload processing unit [*253*] of Claim 26 and further comprising:

(g) a request queue [*251*], operatively coupled to the ingress ports, for receiving respective ingressing, payload-processing requests from the ingress ports and for time-aligning the received payload-processing requests to one another for subsequent arbitration [*252*] within subsequent ones of the local processing windows; and

(h) a request processor [*252*], operatively coupled to the request queue, for arbitrating amongst competing ones of the time-aligned, payload-processing requests, and for issuing grant signals [*261a,514D*] for corresponding ones of the payload-processing requests that win a respective arbitration in a corresponding one of the local processing windows, where each issued grant signal includes a grant time stamp [*585*] identifying a future one [*235o"*] of the local processing windows in which a corresponding payload signal is to be processed by the payloads processing module [*255*].

31. The payload processing unit [*253*] of Claim 30 and further wherein:

(f.1) the post-process output module [*260*] is operatively coupled to the request processor [*252*] and packages, for output during a given one of the local processing windows and by way of a respective egress port [*645-649*], the issued grant signals [*261a,514D*] if any for that respective egress port together with the processed payloads if any, from the payloads processing module.

32. The payload processing unit [*253*] of Claim 30 and further comprising:

(i) a grant identification queue [*249*], coupled to the request processor [*252*] and to the payloads aligning module [*254*], for storing grant identification information [*232c,237c*] indicating for respective ones of the local processing windows in which payload signals may be processed, at least whether a valid grant signal was issued with a corresponding grant time stamp for the respective ones of the local processing windows or at least which of a corresponding plurality of ingressing payload signals is to be acquired from the payloads aligning module [*254*] for processing during the corresponding local processing window.

33. The payload processing unit [*253*] of Claim 30 and further wherein:

(f.1) the post-process output module [*260*] is operatively coupled to the request processor [*252*] and packages, for output during a given one of the local processing windows and by way of a respective egress port [*645-649*], the processed payloads if any, for that respective egress port together with a request back-pressure signal [*146*], which if active, indicates that associated payload-processing requests of that egress port are ingressing at an effective rate greater than a predefined threshold rate.

34. The payload processing unit [*253*] of Claim 30 and further wherein:

(f.1) the post-process output module [*260*] is operatively coupled to the request processor [*252*] and packages, for output during a given one of the local processing windows and by way of a respective egress port [*645-649*], the processed payloads if any, for that respective egress port together with reordering signals [*525,526*] which indicate to a receiver [*680*] of the respective egress port where in a logical order of other payload signals, the processed and packaged payload belongs.

35. The payload processing unit [*253*] of Claim 26 wherein said payloads processing module includes a payload switching matrix [*255'*].